

DT15 Rec'd PCT/PTO 14 JUL 2004

Berlin, 17.06.2004  
GZ: 26 1134-PCT/US/04

Inventor/Applicant:  
Oliver Völckers  
Im Kirschengarten 26  
D-54294 Trier  
Germany

Method for input of text by selection of letters using a  
cursor and device for carrying out said Method

The invention describes a procedure for text input by selecting letters with a cursor visible on a screen that can be used with compact electronic devices that cannot be equipped with a normal typewriter-style keyboard due to lack of space and therefore offer only a secondary text input function with cursor keys or similar input controls; it also describes an apparatus to implement this procedure.

Modern electronic devices with a software operating system and a display often require a text input function but lack a full typewriter-style keyboard. A number of ersatz solutions have been developed that allow text input by means of a selection of letters with a cursor.

The most simple solution consists of a letter set shown on a display, with a cursor that can be controlled in two or four directions with a joystick or with keys. In order to advance the cursor by several positions, the keys have to be pressed several times accordingly. A further developed solution on this basis allows to keep the keys or the

joystick activated for a certain minimum delay, leading to an automatic movement of the cursor in specified time intervals (auto-repeat function).

In order to move the cursor over longer distances, the speed of the cursor movement can be increased gradually (acceleration).

Touch screens, digitizing tablets and sensor strips allow to position a cursor directly at the desired position.

Predictive Text Input can guess a complete input sequence from a reduced information (9 keys instead of 26) on the basis of dictionary. This solution is in common use with mobile phones.

While the solution with two cursor keys is simple to manufacture and easy to learn, it requires numerous key presses or delays for each input character, so it is only accepted as a makeshift solution.

The cursor control with auto-repeat resp. acceleration makes it easier to traverse longer distances. The fast movement, however, leads to the common situation that the cursor surpasses the target so that the cursor position must be corrected tediously. This is because it is impossible to slow down the cursor movement.

While touch screens, digitizing tablets and sensor strips allow the direct selection of a cursor position, the large number of input alternatives leads to errors such as

unintentionally selecting a position directly adjacent to the desired target position.

Predictive text input improves data input with numeric keypads significantly, as long as the input is limited to the vocabulary of its dictionary. In order to allow other input sequences, predictive text input must be turned off.

The known text input functions with a cursor control make use of a cursor with a constant size as well as targets (letter sets) with an identical size. Only predictive text input limits the possible input sequences to those that are assumed on the basis of an electronic dictionary.

Words of natural language contain some redundancy, i.e. after any letter only specific subsequent letters are possible to form meaningful words. For example, in the English and German languages words beginning with „Qua“ often continue with a „d“, „n“ or „r“, but not with an „a“ or „q“.

The well-known and existing text input cursor control systems offer numerous letters for a selection, although these cannot be used for meaningful words at all. On the other side, predictive text input limits input sequences to words in a dictionary, although other letter sequences may be required for name or address information.

A procedure to support text entry by selecting letters with a cursor from a list should function with various cursor

control hardware. It should also allow rapid input without limiting the vocabulary as predictive systems do.

The aim of the invention is to design a system for text entry by selecting letters from a list with a cursor in a user-friendly way, so that users do not have to memorize special input sequences and anybody can use it without special training.

The invention is based on the task to develop a procedure for text input by selecting letters from a list with a cursor that works with a variety of user interface elements for cursor control. Another inventor's task is to develop an apparatus that implements this procedure.

The task is solved as shown in the independent claims.

The invented procedure to enter text by selecting letters from a list of letters with a cursor is based on a selection file that has been generated previously in a way according to the invention. This selection file contains letter sequences with weight assignments that have been produced by means of the mathematical method of frequency statistics. A graphical or audible cursor uses this selection file to offer selections of letters. This procedure is based on the inventive step to utilize the uneven distribution of letters in texts for text entry.

According to the invention, the probability of the occurrence of certain letter combinations is calculated by means of a table, the statistical frequencies are

automatically computed from a typical sample text and made available for the letter selection according to the invention. Depending on the probability of their occurrence in a certain context the cursor appears larger on expected letters than on unlikely letters. This allows to enter letter sequences of common texts faster and with fewer typing errors.

All further characteristics of the invention may be derived from the claims.

The invention shall be described in detail by reference to an example. The accompanying drawings show:

Fig. 1: block diagram of the components of the apparatus according to the invention to implement the procedure

Fig. 2: screen display of a letter list with identical size

Fig. 3: screen display of letter lists with variable sizes

Fig. 4: another screen display of a letter list with variable size

Fig. 5: Flow chart of the procedure's steps during input

Fig. 6: Comparison demonstrating the adjustable effect

Fig. 7: procedure steps to generate a frequency table

Fig. 8: Memory requirements of some frequency tables

Fig. 9: frequency table of a German sample text

Fig. 1 shows a block diagram of the apparatus according to the invention. A cursor control unit 11 is connected to a display 13 via a control unit such as a micro controller 12. The display 13 shows a letter set 14 completely or partially. This letter set 14 contains all characters that can be entered in this situation. The cursor 15 can be moved across the letter set 14 by means of the cursor control unit 11. An indicator 16 formed like a small symbol or mouse pointer within the area of the cursor 15 can optionally deliver a precise feedback to the activation of the cursor control unit 11.

The cursor 15 is displayed on the letter set in variable size depending on the input sequence. To this end, the micro controller 12 stores the last entered letters in a random access memory 18. The micro controller 12 compares this typically incomplete word stored in the random access memory 18 with a catalogue of frequencies of possible input sequences from the read only memory 17.

This comparison allows the micro controller 12 to predict which letters are more likely and which are less likely to follow. Depending on this probability, the contours of the letters of the letter set 14 are highlighted with a variable size when they are selected by means of the cursor 15.

The cursor control unit 11 can in its simplest form just consist of two arrow keys, but it can also consist of a joy stick, a rotary emitter (jog dial), a sensor strip (e.g. linear potentiometer), a touch screen, a mouse etc.. The display 13 must be capable of displaying a subset of the letter set 14 which consists at least of a single character and the cursor 15 and as a maximum the complete repertoire.

It is sufficient if the random access memory 18 has a capacity of approx. 20 characters (20 bytes) und it may lose its content when powered off (volatile memory). The read-only memory 17 needs to have a capacity of at least a few thousand characters (20 to 1000 kilo bytes) in order to store the letter sequence frequencies and it must be non-volatile memory (type ROM or EPROM, hardware storage or firmware).

Fig. 2, 3 and 4 illustrate samples of screen displays of the user interface. Fig. 2 shows a list of letters with a subset from the letter set, all letters shown with identical width. That is equivalent to a cursor control for letter selection according to the state of art (prior art).

The cursor can reach any visible letter theoretically within the same time, although depending on the implementation of the cursor control unit a longer distance may require a longer time. (E.g. in the example of Fig. 2 from the position „b“, the „c“ is faster to access than the „g“, but the difference of the movement from „b“ to „g“ compared to „b“ to „h“ is minimal.)

Fig. 3 clearly shows the implications of the invention. The letters are shown with a variable width, so that in this example the cursor passes more rapidly over the narrowly displayed letters „c“ or „d“ than over the letters „b“ or „g“. The three lines of Fig. 3 show three states that follow each other. The cursor is moved to the right, moving the indicator 16 (introduced in Fig. 1) with a constant speed to the right. Exactly when the indicator touches the edge of the cursor 15 outline, the cursor advances by one letter. This behaviour makes it easier to position the cursor on the widely displayed letters than on the narrowly displayed letters.

Fig. 3 illustrates a situation where the letters „b“, „e“, „g“ and „h“ have been determined as probable following letters, while the letters „a“, „c“ and „d“ have been determined as unlikely. The variable size allows to hit the expected letters faster and easier compared to the less likely letters. In the majority of cases, the larger displayed letters will be selected, because the letter sequences are not randomly distributed in natural language texts. This means the selection is faster and the risk of input errors is reduced.

According to Fitt's Law, targets can be hit with a cursor control faster in proportion to the size of the target. It is realistic to expect that the desired letters can be displayed on average 30 to 60% larger with the invention than without. This allows a rough estimation of an increased input speed by 30 to 60% with the invention.



Fig. 4 displays another situation with the same user interface. After every letter input, the probabilities of the specific following letter are computed again by means of the table 17 of Fig. 1, allowing a corresponding update of the letter set display and their specific sizes.

The figures 3 and 4 show a variable width of letters changing after every input. As an alternative to this, it would be possible to display all letters of the letter set with a constant size as in Fig. 1 and have a variable size cursor, or it would be possible to have letters with variable height or both with variable height and variable width for every letter.

The best solution depends on the type of cursor control and the display: A two-dimensional cursor control (e.g, joystick or mouse) suggests a two-dimensional adaptation, e.g. width and height. If the cursor can only be moved horizontally or vertically, then a variation only along this axis is sensible.

Fig. 5 demonstrates the steps performed during input. After the start 51, the memory with the current input sequence (no. 18 in Fig. 1) is cleared. Then the micro processor 12 computes the probability of occurrence for every letter of the letter set 14 based on the catalogue of input sequence frequencies from the read-only memory 17 (step 53). According to these probabilities, the relative sizes are computed for every displayed letter (step 54) and the display is updated (step 55).

Step 56 means that the cursor movements are handled until the next letter is selected with a key press, mouse click etc. depending on the type of cursor control unit. If the next letter is a space or a word dividing character, the buffer for the input sequence is cleared (step 57 and then 52). However if the word is continued (step 57 and then 58) then the new letter is also memorized in the buffer, so that it can be taken into account upon the next calculation of word probabilities with step 53.

Figure 6 discriminates the invention against known text input cursor control systems and the known procedure of word prediction. According to common cursor control systems for text input, all letters are displayed with an identical sized cursor as shown in Fig. 2 and characterized at the left side of Fig. 6.. By contrast, the word prediction method restricts the selection of letters that are practically assigned either 0% or 100%, as characterized at the right side of Fig. 6..

The effect of the invention, however, may be scaled to any level, as characterized in the middle range of Fig. 6.. Depending on how strong the influence of the letter probabilities to the display is elected, the letter width can vary less (left side of the diagram) or more (right side of the diagram).

Fig. 7 lists the necessary steps to generate the table with the probabilities of the letter sequences. According to the invention, this procedure is performed automatically, if a long and representative text is available that contains

similar sequences as the planned content (step 70). A typical text with a few ten thousand characters is sufficient. The easiest method is to make use of texts that have already been entered by users in practice. If texts for different languages shall be entered, then a separate table should be generated for each language.

First of all, this representative text must be filtered as a preparation for analysis, by converting upper-case letters to lower-case letters and removing all special characters that are irrelevant for the separate words (step 71). By recognizing spaces between words as word separators the text is changed into a word list.

Within this word list, the letter sequences for every word are determined and every occurrence of a sequence is counted (step 72). Depending on the available memory, longer or shorter letter sequences may be considered. In practice, sequences of at least two and no more than six letters should be considered. Longer letter sequences result in more precise statistics and optimized letter cursor sizes.

For example, if the letter sequences in the word „rokoko“ are counted like this, then counting two-letter sequences would deliver „\_r“, „ro“, „ok“, „ko“, „ok“, „ko“, i.e. once „\_r“ and „ro“ and twice „ok“ and „ko“; the character „\_“ indicating a word separator. Counting three-letter sequences would deliver „\_ro“, „rok“, „oko“, „kok“, „oko“, i.e. once „\_ro“, „rok“, „kok“ and twice „oko“.

Finally, step 73 of Fig. 7 symbolizes the calculation of relative probabilities from the statistic counters by setting individual occurrences into relation with the total count. For example, if in the representative text the letter „g” is followed an „r” three times, followed by an „u” once and no other letters, then the probability of a „g” followed by an „r” appears as  $3/4$  and that of a „g” followed by a „u” appears as  $1/4$ .

Fig. 8 lists the required memory for the frequency lists (no. 17 in fig. 1) depending on the length of letter sequences. This is based on an alphabet of 30 letters. Under these conditions, a table with two-letter sequences requires 30 times 30 characters, i.e. 900 bytes. A three-letter sequence table would require accordingly  $30 \times 30 \times 30 = 27,000$  bytes. If tables for several languages shall be stored, the memory requirements multiply accordingly. Fig. 8 demonstrates that the memory requirements for the tables lie in a range that is perfectly acceptable for common electronic devices.

Fig. 9 shows a table with countings of two-letter sequences generated from a 5000 character long sample text in German language. The top line lists each beginning letter and the most left column shows the following letter. This counting reveals for example, that the sample text contained 14 occurrences of the letter sequence „ab” and 4 occurrences of the letter sequence „ba”.

Furthermore, Fig. 9 reveals that the sample text contained no occurrences of „dc” or „dg”, but 61 occurrences of „de”.

If this tendency would be confirmed with a longer representative text, the probability of the sequence „de“ would be considered as high and those of the sequences „dc“ and „dg“ as low. Therefore, after the entry of the letter „d“ the letter set would be displayed in a manner that an „e“ is easily reachable and the letters „c“ and „g“ are not so easy reachable. In other words, an „e“ would then occupy a larger cursor area and „c“ as well as „g“ would only occupy the minimum area.

The difference between minimum and maximum size is defined in a parameter according to Fig. 6.. It is possible to implement this as a user-defined preference, that could perhaps be offered as a choice between 0%, 30%, 60% and 90% (i.e. 90% difference between largest and smallest letter). The size of the complete displayed letter set is constant, only the shares of individual letters are flexible.

Compared to state-of-the-art cursor controls for text input the invention offers a significantly faster input speed. No explanations are required to use it. If the effect is set to a weak level, it might even go unnoticed and will still improve input speed.

By contrast to the word prediction system, the invention does not limit input sequences in any way. Rare letter sequences are only slowed down by means of a smaller representation. The invention is easy to adapt automatically to different languages, as long as a representative text is available. The generated frequency

tables require only minimal space and are scalable to either higher precision or compact size.

The system is adjustable to any display sizes, whether they show the complete alphabet or only a subset of it. In addition, the system can be adapted to all kinds of cursor controls, with arrow keys in two or four directions, joysticks, mice, sensor strips and similar.

The procedure is preferable with compact electronic devices which cannot be equipped with a full typewriter keyboard because of lacking space and therefore contain only a makeshift text input system with cursor keys or similar input devices.

Berlin, 17.06.2004  
GZ: 26 1134-PCT/US/04

### **Definitions**

- 11 Cursor Control Unit
- 12 Micro Controller
- 13 Display
- 14 Letter Set
- 15 Cursor
- 16 Indicator
- 17 Read Only Memory
- 18 Random Access Memory